

Biotic Prediction

Building the Computational Technology Infrastructure
for Public Health and Environmental Forecasting

Software Requirements Document

BP-SRD-1.4

Task Agreement: GSFC-CT-1

August 22, 2003

Contents

1	Overview	3
1.1	Introduction	3
1.2	Referenced Documents	3
1.3	Document Overview	3
2	System Overview	4
2.1	System Concept	4
2.2	System Environment	4
2.3	System Architecture	4
3	Functional Requirements	6
3.1	User Interface	6
3.1.1	Profile Database	6
3.1.2	Roles	6
3.1.3	Graphical User Interface	6
3.1.4	File Management	7
3.2	Ingest	7
3.2.1	Validation	7
3.2.2	Field Data	7
3.2.3	Remote Sensing Data	7
3.2.4	Data Acquisition	7
3.2.5	Monitoring and Reporting	8
3.3	Pre-processing	8
3.3.1	Merge Data	8
3.4	Modeling	8
3.5	Post-processing	9
3.5.1	Reprojecting data	9
3.5.2	Data Overlay	9
3.5.3	Metadata	9
3.6	Archive	9
3.6.1	Database	9
3.6.2	Internal File Store	9
3.6.3	External Files	9
4	Performance Requirements	9
4.1	CT Project Scaling Milestones	9
5	Requirements Notes	11
6	Requirements Verification Matrix	16
7	Build Plan	20
A	Glossary	24

List of Tables

1	Referenced Documents	3
2	Requirements Notes	11
3	Verification Methods	16
4	Requirements Verification Matrix	16
5	Build Plan	20

List of Figures

1	ISFS Functional Architecture	5
---	--	---

1 Overview

1.1 Introduction

This project will develop the high-performance, computational technology infrastructure needed to analyze the past, present, and future geospatial distributions of living components of Earth environments. This involves moving a suite of key predictive, geostatistical biological models into a scalable, cost-effective cluster computing framework; collecting and integrating diverse Earth observational datasets for input into these models; and deploying this functionality as a Web-based service. The resulting infrastructure will be used in the ecological analysis and prediction of exotic species invasions. This new capability will be deployed at the USGS Midcontinent Ecological Science Center and extended to other scientific communities through the USGS National Biological Information Infrastructure program.

1.2 Referenced Documents

Table 1. Referenced Documents

Document Title	Version	Date
Software Engineering / Development Plan	1.2	2002-09-26
Concept of Operations	1.6	2002-10-17

1.3 Document Overview

This document, the *Software Requirements Document*, enumerates the software requirements for the *Invasive Species Forecasting System* (ISFS).

Some of these requirements are planned for immediate implementation in the software, and some are intended to be “design goals” for now and may be implemented in later builds of the software. By enumerating all of the requirements here, the software is described in its eventual, evolved state.

Section 2 provides a brief description of the system, its interfaces, and its functional subsystems. Sections 3, 4 and ?? lay out the explicit list of Functional, Performance, and Documentation Requirements. Further revisions of this document will break down these requirements into more detail and add more specific derived requirements as needed.

Section 6 provides a verification matrix with the method that will be used to verify that a requirement has been adequately attained. This matrix will be used as the basis for the *Software Test Plan* to be delivered later.

The Build Plan Matrix in Section 7 will lay out the time frame for the implementation of each requirement.

2 System Overview

2.1 System Concept

The ISFS will provide an environment for the application of different statistical models to geolocated reference datasets.

Users will be able to:

- Apply pre-defined models against existing datasets
- Add new or augment existing datasets
- Add new or alter existing models.

A primary goal of the evolved system is to optimize resource intensive computation by distributing it across a Linux cluster. This will minimize the time it takes to run the models and provide a better user experience.

In order to provide a baseline for comparison, the canonical system will:

- Show proof of concept for the canonical example and the validity of the example
- Establish metrics that can gauge performance of the system as a whole
- Provide an established baseline configuration from which future development can be compared
- Identify specific targets where computational processes can be enhanced to show scalar improvements in performance
- Identify the algorithms and tools that are used for processing and document these for the user.

2.2 System Environment

The *Invasive Species Forecasting System* will, for this stage of development be hosted on a Linux server at the NASA Goddard Space Flight Center, Building 28. It will use COTS tools including ENVI image processing, IDL, Fortran programs, and shell scripting languages. Datasets are already loaded and in the correct formats for processing and benchmarking the canonical example.

The establishment of a dedicated server will allow comparison of performance and product quality as different operating parameters are modified. Once the benchmarks for the baseline system have been recorded; the exact configuration of the dedicated server will be archived. Then, the server will be used to prototype different operational scenarios as documented in the BP-CONOP document.

Subsequent mods to the baseline will be scrupulously documented along with any changes in performance that are measured or observed.

2.3 System Architecture

The ISFS conceptually consists of 3 layers comprised of 6 functional elements as shown in Figure 1:

Front-End Layer:

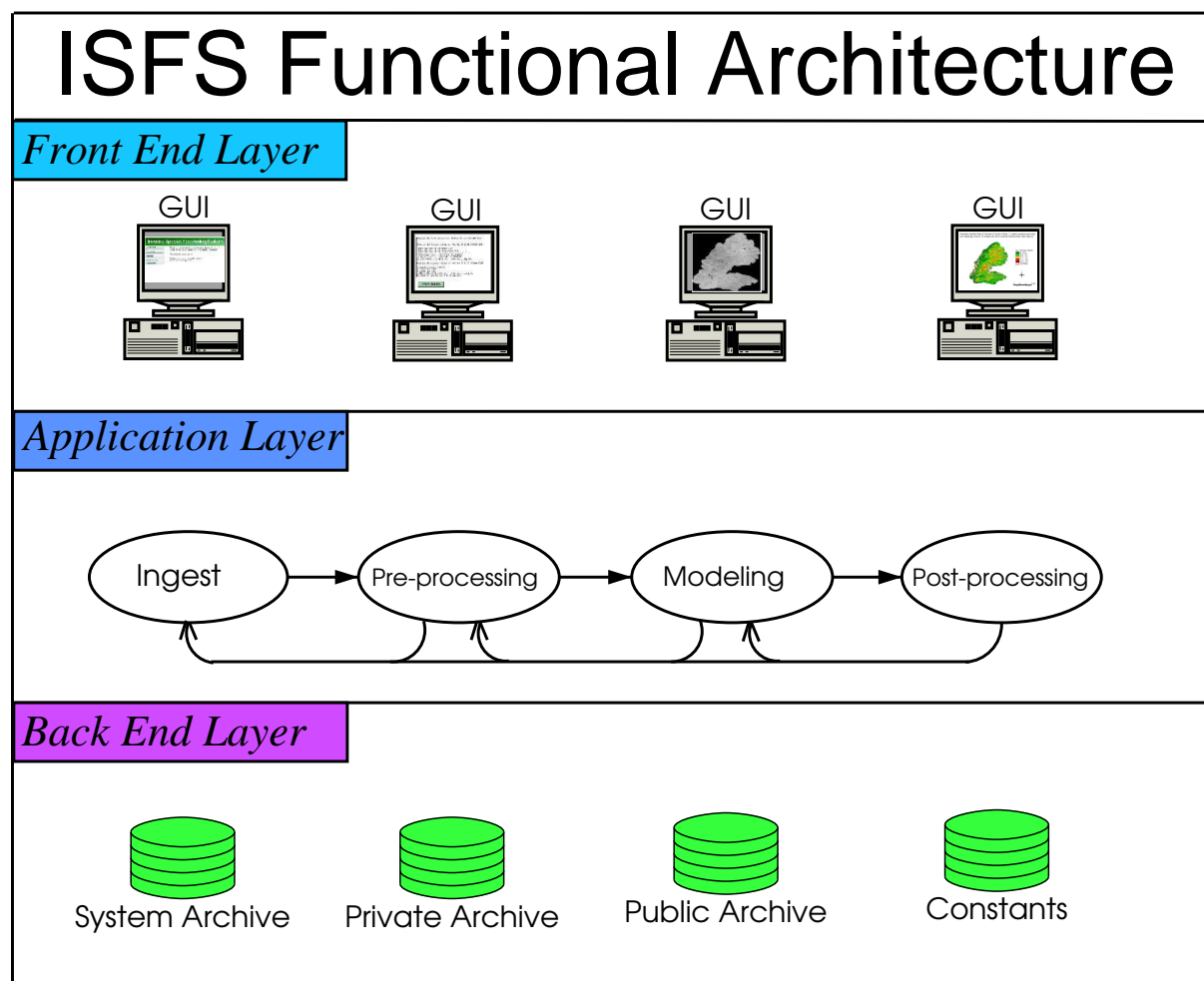
1. User Interface — A web presence is maintained offering users universal access to the system.

Application Layer:

2. Ingest — The ingest subsystem will serve as the initial "entry point" for all data used in the system, offering authorized users the capability of uploading data to be used by the system.
3. Pre-processing — Pre-processing activities merge ingested datasets to create a data product that can be analyzed in the modeling step.
4. Modeling — Modeling will predict a certain species migration through or invasion of habitat based on remote sensing imagery and ancillary data layers.
5. Post-processing — The post-processing step applies the results of the modeling activities to generate products in the form of images and corresponding data for user analysis.

Backend Layer:

6. Archive — An archive subsystem will be coordinated by a database that will store pointers to archived files.

**Figure 1.** ISFS Functional Architecture

3 Functional Requirements

3.1 User Interface

3.1.1 Profile Database

3.1.1.1 The system shall maintain a profile database of users with:

1. User's status
2. User's preferences
3. User's role

3.1.2 Roles

3.1.2.1 The system shall support various roles that control access to various capabilities of the system:

1. "Administrator" — This role provides complete access to the ISFS web system. An administrator would be able to manage other users and have access to their workspace, preferences and authorization information.
2. "Model Builder" — This role is intended for the user that wishes to configure and tailor the model related tasks. The Model Builder is someone who has been authenticated and authorized to upload and ingest data in preparation for other users to perform repeated model runs. The Model Builder has registered with the system and maintains an active logon name and password to access the system. The Model Builder maintains a profile within the system that remembers the Builder's data selections and makes those selections available to the Builder upon login. The Model Builder does not have complete access to the ISFS web system, as complete access is the role of the administrator.
3. "Model User" — This role is intended for a user to select specific input data and run it through a previously configured ISFS model run. The Model User has registered with the system and maintains an active logon name and password to access the system. The Model User maintains a profile within the system that remembers the User's data selections and makes those selections available to the User upon login. The Model Builder is someone who has not been authorized to upload and ingest data to the system.

3.1.3 Graphical User Interface

3.1.3.1 The system shall include a Graphical User Interface ("GUI") to support user interaction with the system.

3.1.3.2 The GUI shall dynamically construct personalized web pages based on the Profile Database and the User's Role.

3.1.3.3 The GUI shall display predictive map and uncertainty map output.

3.1.3.4 The GUI shall invite all system users to register in order to use the system based on their roles.

3.1.3.5 The system shall allow the model builder to create new models.

3.1.3.6 The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.

3.1.4 File Management

3.1.4.1 The user shall have the option of saving run results with annotations in personal repository.

3.2 Ingest

3.2.1 Validation

3.2.1.1 The system shall verify integrity, but is not required to validate the quality of the data before ingest.

3.2.1.2 The system will accept data from an authoritative source. An authoritative source is someone who has registered with the system and been authorized to upload data to the system.

3.2.1.3 The system shall log all data sources.

3.2.2 Field Data

3.2.2.1 The system shall provide standard templates for ingesting field data in a tabular form.

3.2.2.2 The templates shall include all required fields to be captured.

3.2.2.3 The templates shall be in an accessible format (such as spreadsheet, database, or simple ASCII list). The format will include tab and white space delimited tables along with comma delimited.

3.2.3 Remote Sensing Data

3.2.3.1 The system shall support ingest from external satellite data archives (such as the Goddard DAAC).

3.2.3.2 The system shall support ingest of user-supplied satellite data or airborne imagery from digital files.

3.2.3.3 The system shall support ingest of user-supplied data, also known as user-supplied layers.

3.2.3.4 The system shall support ingest of user-supplied data files for ancillary layers. Ancillary layers consist of descriptive comments and other ancillary data that are relevant to the user-supplied data.

3.2.4 Data Acquisition

3.2.4.1 The system shall support secured ftp-push for ingest of user supplied data.

3.2.4.2 The system shall support automated secured ftp pull from external archives.

3.2.4.3 The system shall provide accounting and logging by requiring users' name and password.

3.2.4.4 The system shall maintain a list of external archives and required data sets and these shall be available to all users of the system.

3.2.4.5 The interface between the system and each external archive will be thoroughly documented as those connections are tested and verified.

3.2.5 Monitoring and Reporting

- 3.2.5.1 The ingest subsystem shall monitor the number and volume of data brought into the system and this information will be recorded.
- 3.2.5.2 The ingest subsystem shall produce data reports broken down by location, user, and external archive.
- 3.2.5.3 The system shall generate and display associated metadata describing output files, runtime parameters, and performance statistics.

3.3 Pre-processing

3.3.1 Merge Data

- 3.3.1.1 The system shall merge ingested datasets into a data product that can be analyzed by the modeling subsystem.
- 3.3.1.2 The system shall perform resampling if the input data are not at the same resolution.
- 3.3.1.3 The data shall be converted to a common analysis format.

3.4 Modeling

- 3.4.1 The system shall provide the capability to specify response (dependent “Y variable”) and explanatory (independent “X variable”) variables from available databases.
- 3.4.2 The system shall provide graphical techniques (selection by rectangular or circular region) to explore the relationships between these variables.
- 3.4.3 The system shall have the ability to “fit” models through least squares (or other) optimization routines.
- 3.4.4 The system shall provide screening techniques to quantitatively assess which explanatory variables are related to the response variable (such as stepwise regression).
- 3.4.5 The system shall calculate geospatial statistics (such as variograms).
- 3.4.6 The system shall incorporate spatial structure into the modeling (such as generalized least squares or kriging).
- 3.4.7 The system shall be able to output model results and relevant model diagnostics.
- 3.4.8 The system shall allow the model builder to create new models.
- 3.4.9 The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.

3.5 Post-processing

3.5.1 Reprojecting data

3.5.1.1 The system shall display an image of the output data.

3.5.1.2 The system shall produce a data file suitable for reprojection by external COTS utilities.

3.5.2 Data Overlay

3.5.2.1 The system shall overlay output data with other layers as requested.

3.5.3 Metadata

3.5.3.1 Output data shall be packaged with appropriate metadata.

3.5.3.2 Each output data set shall be assigned a unique identifier.

3.6 Archive

3.6.1 Database

3.6.1.1 The Archive shall have a database that will store pointers to the archived files.

3.6.1.2 The database shall be able to refer to “internal” (stored in the local **File Store**) and “external” files.

3.6.1.3 All files (“internal” and “external”) shall be indexed with a unique file ID.

3.6.2 Internal File Store

3.6.2.1 Files shall be stored in a logically arranged directory structure.

3.6.3 External Files

3.6.3.1 For external files, the archive system shall store a pointer or URL that can be used to retrieve and stage the files for subsequent processing.

4 Performance Requirements

4.1 CT Project Scaling Milestones

4.1.1 Improve implementation of *PlantDiversity* to deliver canonical products 25X faster than the baseline implementation.

4.1.2 Improve implementation of *PlantDiversity* to accomodate 10X more sample data at 25X the time required in the baseline implementation and 10X larger area at 2.5X the time required in the baseline implementation.

- 4.1.3 Improve implementation of *PlantDiversity* to achieve the advanced application goals of delivering the canonical products 200X faster than the baseline implementation on a 256-node cluster and accomodating 100X more sample data 1000X faster than the baesline and 100X larger area 10X faster than the baseline on a 1024-node cluster.

5 Requirements Notes

Table 2. Requirements Notes

ID	Requirement	Source	Dependencies	Notes
3.	Functional Requirements			
3.1	User Interface			
3.1.1	Profile Database			
3.1.1.1	The system shall maintain a profile database of users.			
3.1.2	Roles			
3.1.2.1	The system shall support various roles that control access to various capabilities of the system.			
3.1.3	Graphical User Interface			
3.1.3.1	The system shall include a Graphical User Interface (GUI) to support user interaction with the system.			
3.1.3.2	The GUI shall dynamically construct personalized web pages based on the Profile Database and the User s Role.			
3.1.3.3	The GUI shall display predictive map and uncertainty map output.			
3.1.3.4	The GUI shall invite all system users to register in order to use the system based on their roles.			
3.1.3.5	The system shall allow the model builder to create new models.			
3.1.3.6	The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.			
3.1.4	File Management			
3.1.4.1	The user shall have the option of saving run results with annotations in personal repository.			
3.2	Ingest			
3.2.1	Validation			

ID	Requirement	Source	Dependencies	Notes
3.2.1.1	The system shall verify integrity, but is not required to validate the quality of the data before ingest.			
3.2.1.2	The system will accept data from an authoritative source.			
3.2.1.3	The system shall log all data sources.			
3.2.2	Field Data			
3.2.2.1	The system shall provide standard templates for ingesting field data in a tabular form.			
3.2.2.2	The templates shall include all required fields to be captured.			
3.2.2.3	The templates shall be in an accessible format.			
3.2.3	Remote Sensing Data			
3.2.3.1	The system shall support ingest from external satellite data archives.			
3.2.3.2	The system shall support ingest of user-supplied satellite data or airborne imagery from digital files.			
3.2.3.3	The system shall support ingest of user-supplied data.			
3.2.3.4	The system shall support ingest of user-supplied data files for ancillary layers.			
3.2.4	Data Acquisition			
3.2.4.1	The system shall support secured ftp-push for ingest of user supplied data.			
3.2.4.2	The system shall support automated secured ftp pull from external archives.			
3.2.4.3	The system shall provide accounting and logging by requiring users name and password.			
3.2.4.4	The system shall maintain a list of external archives and required data sets.			

ID	Requirement	Source	Dependencies	Notes
3.2.4.5	The interface between the system and each external archive will be thoroughly documented.			
3.2.5	Monitoring and Reporting			
3.2.5.1	The ingest subsystem shall monitor the number and volume of data brought into the system.			
3.2.5.2	The ingest subsystem shall produce data reports broken down by location, user, and external archive.			
3.2.5.3	The system shall generate and display associated metadata describing output files, runtime parameters, and performance statistics.			
3.3	Pre-processing			
3.3.1	Merge Data			
3.3.1.1	The system shall merge ingested datasets.			
3.3.1.2	The system shall perform resampling if the input data are not at the same resolution.			
3.3.1.3	The data shall be converted to a common analysis format.			
3.4	Modeling			
3.4.1	The system shall allow the ability to specify response and explanatory variables from the available databases.			
3.4.2	The system shall provide graphical techniques to explore the relationships between these variables.			
3.4.3	The system shall have the ability to fit models through least squares (or other) optimization routines.			
3.4.4	The system shall provide screening techniques to quantitatively assess which explanatory variables are related to the response variable.			

ID	Requirement	Source	Dependencies	Notes
3.4.5	The system shall calculate geospatial statistics.			
3.4.6	The system shall incorporate spatial structure into the modeling.			
3.4.7	The system shall be able to output model results and relevant model diagnostics.			
3.4.8	The system shall allow the model builder to create new models.			
3.4.9	The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.			
3.5	Post-processing			
3.5.1	Reprojecting data			
3.5.1.1	The system shall display an image of the output data.			
3.5.1.2	The system shall produce a data file suitable for reprojection by external COTS utilities.			
3.5.2	Data Overlay			
3.5.2.1	The system shall overlay output data with other layers as requested.			
3.5.3	Metadata			
3.5.3.1	Output data shall be packaged with appropriate metadata.			
3.5.3.2	Each output data set shall be assigned a unique identifier.			
3.6	Archive			
3.6.1	Database			
3.6.1.1	The Archive shall have a database that will store pointers to the archived files.			
3.6.1.2	The database shall be able to refer to internal (stored in the local File Store) and external files.			
3.6.1.3	All files (internal and external) shall be indexed with a unique file ID.			

ID	Requirement	Source	Dependencies	Notes
3.6.2	Internal File Store			
3.6.2.1	Files shall be stored in a logically arranged directory structure.			
3.6.3	External Files			
3.6.3.1	For external files, the archive system shall store a pointer that can be used to retrieve and stage the files for subsequent processing.			
4.	Performance Requirements			
4.1	CT Project Scaling Milestones			
4.1.1	Deliver canonical products 25X faster than the baseline implementation			
4.1.2	Accommodate 10X more sample data at 25X the time required in the baseline implementation and 10X larger area at 2.5X the time required in the baseline implementation.			
4.1.3	Deliver canonical products 200X faster than the baseline implementation on a 256-node cluster accomodating 100X more sample data 1000X faster than the baseline and 100X larger area 10X faster than the baseline on a 1024-node cluster.			

6 Requirements Verification Matrix

Table 3. Verification Methods

Verification Method	Description
Demonstration	The operation of the system, or a part of the system that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
Test	The operation of the system, or a part of the system, using instrumentation or other special test equipment to collect data for later analysis.
Analysis	The processing of accumulated data obtained from other qualification methods. Examples are reduction, interpolation, or extrapolation of test results.
Inspection	The visual examination of system components, documentation, etc.

Table 4. Requirements Verification Matrix

ID	Requirement	Ver. Meth.
3.	Functional Requirements	
3.1	User Interface	
3.1.1	Profile Database	
3.1.1.1	The system shall maintain a profile database of users.	Demo.
3.1.2	Roles	
3.1.2.1	The system shall support various roles that control access to various capabilities of the system.	Demo.
3.1.3	Graphical User Interface	
3.1.3.1	The system shall include a Graphical User Interface (GUI) to support user interaction with the system.	Demo.
3.1.3.2	The GUI shall dynamically construct personalized web pages based on the Profile Database and the User's Role.	Demo.
3.1.3.3	The GUI shall display predictive map and uncertainty map output.	Demo.
3.1.3.4	The GUI shall invite all system users to register in order to use the system based on their roles.	Demo.
3.1.3.5	The system shall allow the model builder to create new models.	Demo.
3.1.3.6	The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.	Demo.
3.1.4		
3.1.4.1	The user shall have the option of saving run results with annotations in personal repository.	Demo.
3.2	Ingest	
3.2.1	Validation	
3.2.1.1	The system shall verify integrity, but is not required to validate the quality of the data before ingest.	Test

ID	Requirement	Ver. Meth.
3.2.1.2	The system will accept data from an authoritative source.	Demo./Anal.
3.2.1.3	The system shall log all data sources.	Demo.
3.2.2	Field Data	
3.2.2.1	The system shall provide standard templates for ingesting field data in a tabular form.	Demo.
3.2.2.2	The templates shall include all required fields to be captured.	Demo.
3.2.2.3	The templates shall be in an accessible format.	Demo./Anal.
3.2.3	Remote Sensing Data	
3.2.3.1	The system shall support ingest from external satellite data archives.	
3.2.3.2	The system shall support ingest of user-supplied satellite data or airborne imagery from digital files.	Demo.
3.2.3.3	The system shall support ingest of user-supplied data.	Demo.
3.2.3.4	The system shall support ingest of user-supplied data files for ancillary layers.	Demo.
3.2.4	Data Acquisition	
3.2.4.1	The system shall support secured ftp-push for ingest of user supplied data.	Demo.
3.2.4.2	The system shall support automated secured ftp pull from external archives.	Demo.
3.2.4.3	The system shall provide accounting and logging by requiring users name and password.	Demo.
3.2.4.4	The system shall maintain a list of external archives and required data sets.	Demo.
3.2.4.5	The interface between the system and each external archive will be thoroughly documented.	Insp.
3.2.5	Monitoring and Reporting	
3.2.5.1	The ingest subsystem shall monitor the number and volume of data brought into the system.	Test
3.2.5.2	The ingest subsystem shall produce data reports broken down by location, user, and external archive.	Test
3.2.5.3	The system shall generate and display associated metadata describing output files, runtime parameters, and performance statistics.	Demo./Test
3.3	Pre-processing	
3.3.1	Merge Data	
3.3.1.1	The system shall merge ingested datasets.	Demo.
3.3.1.2	The system shall perform resampling if the input data are not at the same resolution.	Demo./Anal.
3.3.1.3	The data shall be converted to a common analysis format.	Demo.
3.4	Modeling	
3.4.1	The system shall allow the ability to specify response and explanatory variables from the available databases.	Demo.

ID	Requirement	Ver. Meth.
3.4.2	The system shall provide graphical techniques to explore the relationships between these variables.	Demo.
3.4.3	The system shall have the ability to fit models through least squares (or other) optimization routines.	Demo./Test/Anal.
3.4.4	The system shall provide screening techniques to quantitatively assess which explanatory variables are related to the response variable.	Demo./Test/Anal.
3.4.5	The system shall calculate geospatial statistics.	Demo./Test/Anal.
3.4.6	The system shall incorporate spatial structure into the modeling.	Demo./Test/Anal.
3.4.7	The system shall be able to output model results and relevant model diagnostics.	Dmo.
3.4.8	The system shall allow the model builder to create new models.	Demo.
3.4.9	The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.	Demo.
3.5	Post-processing	
3.5.1	Reprojecting data	
3.5.1.1	The system shall display an image of the output data.	Demo.
3.5.1.2	The system shall produce a data file suitable for reprojection by external COTS utilities.	Demo.
3.5.2	Data Overlay	
3.5.2.1	The system shall overlay output data with other layers as requested.	Demo.
3.5.3	Metadata	
3.5.3.1	Output data shall be packaged with appropriate metadata.	Demo.
3.5.3.2	Each output data set shall be assigned a unique identifier.	Demo.
3.6	Archive	
3.6.1	Database	
3.6.1.1	The Archive shall have a database that will store pointers to the archived files.	Demo.
3.6.1.2	The database shall be able to refer to internal (stored in the local File Store) and external files.	Demo.
3.6.1.3	All files (internal and external) shall be indexed with a unique file ID.	Demo.
3.6.2	Internal File Store	
3.6.2.1	Files shall be stored in a logically arranged directory structure.	Insp.
3.6.3	External Files	
3.6.3.1	For external files, the archive system shall store a pointer that can be used to retrieve and stage the files for subsequent processing.	Demo.
4.	Performance Requirements	
4.1	CT Project Scaling Milestones	
4.1.1	Deliver canonical products 25X faster than the baseline implementation	Test/Anal.
4.1.2	Accommodate 10X more sample data at 25X the time required in the baseline implementation and 10X larger area at 2.5X the time required in the baseline implementation.	Test/Anal.

ID	Requirement	Ver. Meth.
4.1.3	Deliver canonical products 200X faster than the baseline implementation on a 256-node cluster accomodating 100X more sample data 1000X faster than the baseline and 100X larger area 10X faster than the baseline on a 1024-node cluster.	Test/Anal.

7 Build Plan

- B1** Build 1, Milestone F
B2 Build 2, Milestone G
B3 Build 3, Milestone K
DG Design Goals, future builds

Table 5. Build Plan

ID	Requirement	B1 (F)	B2 (G)	B3 (K)	DG
3.	Functional Requirements				
3.1	User Interface				
3.1.1	Profile Database				
3.1.1.1	The system shall maintain a profile database of users.		X	X	
3.1.2	Roles				
3.1.2.1	The system shall support various roles that control access to various capabilities of the system.		X	X	
3.1.3	Graphical User Interface				
3.1.3.1	The system shall include a Graphical User Interface (GUI) to support user interaction with the system.		X	X	
3.1.3.2	The GUI shall dynamically construct personalized web pages based on the Profile Database and the User's Role.				X
3.1.3.3	The GUI shall display predictive map and uncertainty map output.		X	X	
3.1.3.4	The GUI shall invite all system users to register in order to use the system based on their roles.		X	X	
3.1.3.5	The system shall allow the model builder to create new models.		X	X	
3.1.3.6	The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.		X	X	
3.1.4	File Management				
3.1.4.1	The user shall have the option of saving run results with annotations in personal repository.				X
3.2	Ingest				
3.2.1	Validation				
3.2.1.1	The system shall verify integrity, but is not required to validate the quality of the data before ingest.	X	X	X	
3.2.1.2	The system will accept data from an authoritative source.				X
3.2.1.3	The system shall log all data sources.		X	X	
3.2.2	Field Data				

ID	Requirement	B1 (F)	B2 (G)	B3 (K)	DG
3.2.2.1	The system shall provide standard templates for ingesting field data in a tabular form.		X	X	
3.2.2.2	The templates shall include all required fields to be captured.		X	X	
3.2.2.3	The templates shall be in an accessible format.		X	X	
3.2.3	Remote Sensing Data				
3.2.3.1	The system shall support ingest from external satellite data archives.		X		X
3.2.3.2	The system shall support ingest of user-supplied satellite data or airborne imagery from digital files.		X		
3.2.3.3	The system shall support ingest of user-supplied data.	X	X	X	
3.2.3.4	The system shall support ingest of user-supplied data files for ancillary layers.		X	X	
3.2.4	Data Acquisition				
3.2.4.1	The system shall support secured ftp-push for ingest of user supplied data.		X	X	
3.2.4.2	The system shall support automated secured ftp pull from external archives.				X
3.2.4.3	The system shall provide accounting and logging by requiring users name and password.		X	X	
3.2.4.4	The system shall maintain a list of external archives and required data sets.				X
3.2.4.5	The interface between the system and each external archive will be thoroughly documented.				X
3.2.5	Monitoring and Reporting				
3.2.5.1	The ingest subsystem shall monitor the number and volume of data brought into the system.		X	X	
3.2.5.2	The ingest subsystem shall produce data reports broken down by location, user, and external archive.			X	
3.2.5.3	The system shall generate and display associated metadata describing output files, runtime parameters, and performance statistics.			X	
3.3	Pre-processing				
3.3.1	Merge Data				
3.3.1.1	The system shall merge ingested datasets.				X
3.3.1.2	The system shall perform resampling if the input data are not at the same resolution.				X
3.3.1.3	The data shall be converted to a common analysis format.		X	X	
3.4	Modeling				
3.4.1	The system shall allow the ability to specify response and explanatory variables from the available databases.		X	X	

ID	Requirement	B1 (F)	B2 (G)	B3 (K)	DG
3.4.2	The system shall provide graphical techniques to explore the relationships between these variables.		X	X	
3.4.3	The system shall have the ability to fit models through least squares (or other) optimization routines.	X	X	X	
3.4.4	The system shall provide screening techniques to quantitatively assess which explanatory variables are related to the response variable.		X	X	
3.4.5	The system shall calculate geospatial statistics.	X	X	X	
3.4.6	The system shall incorporate spatial structure into the modeling.				X
3.4.7	The system shall be able to output model results and relevant model diagnostics.	X	X	X	
3.4.8	The system shall allow the model builder to create new models.		X	X	
3.4.9	The system shall allow the model user to select from an assortment of modeling techniques and to modify model parameters.		X	X	
3.5	Post-processing				
3.5.1	Reprojecting data				
3.5.1.1	The system shall display an image of the output data.	X	X	X	
3.5.1.2	The system shall produce a data file suitable for reprojection by external COTS utilities.	X	X	X	
3.5.2	Data Overlay				
3.5.2.1	The system shall overlay output data with other layers as requested.			X	
3.5.3	Metadata				
3.5.3.1	Output data shall be packaged with appropriate metadata.	X	X	X	
3.5.3.2	Each output data set shall be assigned a unique identifier.	X	X	X	
3.6	Archive				
3.6.1	Database				
3.6.1.1	The Archive shall have a database that will store pointers to the archived files.		X	X	
3.6.1.2	The database shall be able to refer to internal (stored in the local File Store) and external files.		X	X	
3.6.1.3	All files (internal and external) shall be indexed with a unique file ID.		X	X	
3.6.2	Internal File Store				
3.6.2.1	Files shall be stored in a logically arranged directory structure.		X	X	

ID	Requirement	B1 (F)	B2 (G)	B3 (K)	DG
3.6.3	External Files				
3.6.3.1	For external files, the archive system shall store a pointer that can be used to retrieve and stage the files for subsequent processing.		X	X	
4.	Performance Requirements				
4.1	CT Project Scaling Milestones				
4.1.1	Deliver canonical products 25X faster than the baseline implementation	X			
4.1.2	Accommodate 10X more sample data at 25X the time required in the baseline implementation and 10X larger area at 2.5X the time required in the baseline implementation.		X		
4.1.3	Deliver canonical products 200X faster than the baseline implementation on a 256-node cluster accomodating 100X more sample data 1000X faster than the baseline and 100X larger area 10X faster than the baseline on a 1024-node cluster.		X		

A Glossary

BP Biotic Prediction project
CT Computational Technologies project
CONOP Concept of Operations
COTS Commercial Off The Shelf
CSU Colorado State University
ESTO Earth Science Technology Office
GSFC Goddard Space Flight Center
GUI Graphical User Interface
ISFS Invasive Species Forecasting System
NREL Natural Resources Ecology Laboratory
SEP Software Engineering / Development Plan
URL Uniform Resource Locator